

*Supervisors:* Rune Lyngsoe, Jotun Hein, Bernard Sufrin

*Project:* Parallelisation of Recombination Analysis

Usually when we think about the evolution of life and speciation, we have in mind a tree like structure. Two groups of a species eventually become so distant that they become different species, creating a new bifurcation in the history of evolution. However, this view completely ignores recombination where a new individual is created by combining genetic material from two other individuals. Including recombination requires evolution to be described by a type of directed acyclic graph, called an *ancestral recombination graph* (ARG), rather than a tree. The most prominent case of recombination is probably in sexual reproduction, where the progeny inherits one chromosome in each of its chromosome pairs from each of its parents. The chromosomes inherited, however, are not exact copies of one of the parental chromosomes. Rather it is a mosaic of the parent in question's pair of that chromosome, created by recombination or cross-over events switching the copying back and forth between the two chromosomes of the pair. In fact, mechanisms exist to discard chromosomes where no recombination has taken place. But meiotic recombination, as it is called, is not the only incidence of recombination. It is believed that all organisms to a smaller or larger extent exchange genetic material by recombination. This includes viruses and bacteria, and recombination is believed to be a major factor in e.g. the emergence of multi-resistant pathogens like MRSA and new strains of influenza.

In our group, we have developed software currently the most powerful in existence to infer recombinations by parsimony under the infinite sites model of substitutions. A first challenge of this project would probably be to figure out what the previous sentence means. To give a brief explanation, assume that we have sequenced the same gene from a number of individuals. Though the sequences will be mostly identical, there will be some bases that differs between any two individuals. Positions where not all individuals have the same base are called *segregating sites*. The infinite sites model of substitutions essentially means that each segregating site can be traced back to one particular point in time. At this point, a mutation substituted the original base in the segregating site with a different base in some individual (making it segregating). This mutation was then passed on and multiplied by inheritance to eventually appear in some of the individuals we sequenced. It is easy to prove, that under this assumption not all data sets allow a simple tree like description of their evolutionary history. Our software finds the minimum number of recombinations that are required in any evolutionary history of the input data, and an accompanying evolutionary history with this number of recombinations. Inferring evolutionary histories by finding one requiring a minimum number of events is known as parsimony inference.

Though our software is the most powerful software available for this type of recombination inference, it is still limited to data sets that can at best be said to be medium sized. Currently the software is limited to sequential execution, so one straightforward improvement would be to add parallelisation to the code. The method implemented essentially explores all possible evolutionary histories by searching back in time from the sampled data to the most recent common ancestor. This means that most of the computation is an embarrassingly parallel branching process. However, there is communication between these branching processes through a hash table. This hash table stores configurations encountered, to avoid searching back from the same configuration more than once. Though profiling indicates that the hash table operations constitutes only about 1% of the CPU usage, the hash table is the major culprit in terms of memory usage of the program. Memory usage will in most cases be the limiting factor. Distributing the hash table therefore will be the most critical part of this parallelisation exercise. Our software is implemented in C, so familiarity with C is a prerequisite for doing this project.

### *Suggested reading*

Beagle homepage at [www.stats.ox.ac.uk/~lyngsoe/beagle/](http://www.stats.ox.ac.uk/~lyngsoe/beagle/)

Gene Genealogies, Variation and Evolution by J. Hein, M.H. Schierup, and C. Wiuf, Oxford University Press (2005)

Minimum Recombination Histories by Branch and Bound by R.B. Lyngsø, Y.S. Song, and J. Hein, proceedings of the 5th Workshop on Algorithms in Bioinformatics (2005)